# Monitoring Students Actions During Examination Using Neural Networks

MSc Research Project
Data Analytics

Bharath Potla
Student ID: X22237534

School of Computing
National College of Ireland

Supervisor: Eamon Nolan

**National College of Ireland**

**MSc Project Submission Sheet**

**School of Computing**

| | | | |
|---|---|---|---|
| **Student Name:** | Bharath Potla | | |
| **Student ID:** | X22237534 | | |
| **Programme:** | MSc in Data Analytics | **Year:** | Jan 2024-Jan 2025 |
| **Module:** | MSc Research Project | | |
| **Supervisor:** | Eamon Nolan | | |
| **Submission Due Date:** | 12 December 2024 | | |
| **Project Title:** | Monitoring Students Actions During Examination Using Neural Networks. | | |
| **Word Count:** | **6488** | **Page Count:** | **19** |

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

**Signature:** Bharath Potla

**Date:** 11-12-2024

**PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST**

| | |
|---|---|
| Attach a completed copy of this sheet to each project (including multiple copies) | □ |
| **Attach a Moodle submission receipt of the online project submission,** to each project (including multiple copies). | □ |
| **You must ensure that you retain a HARD COPY of the project**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. | □ |

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

| **Office Use Only** | |
|---|---|
| Signature: | |
| Date: | |
| Penalty Applied (if applicable): | |

# Monitoring Students Actions During Examination Using Neural Networks

Bharath Potla
X22237534

**Abstract**

In the event where the number of students enrolling into the education institutions increases, cases of academic violation also a case in point is the examinations. The research we introduce a computer vision system for observing student exam behavior by analyzing live video feeds for suspicious activity. The system tracks key facial movements such as eye shifts, lip movements, and head posture with the intent of identifying warning signs of cheating. In order to achieve this goal, this study extends the utility of Python's segmentation and recognition libraries such as OpenCV and TensorFlow by applying it to face recognition to detect subtle facial movements associated with suspicious behavior. Moreover, an object detection module, YOLO (You Only Look Once), is also added into the research to detect banned objects including mobile phones inside examination environment. The system combines facial analysis with object detection and produces a complete solution for surveillance during exams to aid in the detection of dishonest practices in real time. The findings show that the proposed system can accurately capture and analyze pupil behaviors for use as a reliable and automated application to support academic integrity during examine situations.

## 1 Introduction

Monitoring students during exams is a location intensive task and may not always be easy in larger settings. In the past, roles of invigilators have been to oversee the integrity of exam integrity. However, the rise in students taking exams, specifically in online environments post COVID, makes it difficult to monitor effectively. But exams moving to a digital platform has offered students new routes to cheat, especially with mobile devices. Almost all educational institutions relied on utilization of online learning to maintain the teaching and learning schedule for all students. According to UNESCO (2020), 138 countries with 1.37 billion students around the world were taking online classes from home. Currently, existing online instruction environments lack the ability to withhold the learners from scamming the online exams, nor can they ensure fairness and equality for all examinees like those in traditional classroom exams (Jia, J. and He, Y., 2022). While online exams are common, this is still a critical problem of tracking and preventing such behavior with little manpower.

The risk of students cheating for instance, searching for answers online or using forbidden devices like smartphones and tablets is growing among students in the context of high stakes exams especially in educational settings. It is very likely that cheating happens due to the reliance on a small number of invigilators, to the point where in some countries there have even been false accusations because of interpersonal biases. With smart classroom and digital examination becoming widely employed, we need new solutions to maintain academic integrity in a much more efficient and scalable way.

This work will offer an approach for monitoring examinations for suspicious behaviors in real time using computer vision and employing modern deep learning models and frameworks to provide robust and accurate monitoring. ResNet10-SSD is a state of the art convolutional neural network model implemented using OpenCV's DNN module, using which face detection on the video feed is carried out in a very efficient manner guaranteeing high precision in detecting faces (Hettiarachchi, S., 2021). After finding the face, It uses TensorFlow to find and track facial landmarks like eyes, lips, and head placement. It enables the system to look at certain behaviors, for example if someone is talking this is an indicator of suspicious activity (Farkhod, A., Abdusalomov, A.B., Mukhiddinov, M. and Cho, Y.I., 2022). The application uses YOLO for identifying banned objects, as it is capable of detecting multiple objects in real time with high accuracy so that you don't miss banished device types. Preprocessing techniques such as noise reduction and grayscale conversion are used to enhance the system's performance, by enhancing image clarity and thus more efficiently carrying out feature extractions. Furthermore, these tools are implemented as an integrated part of the system, making them capable of adaptation to different examination environments, online or offline, so that manual invigilation can also be replaced with a scalable and automated alternative. However, this method greatly minimizes human bias and errors, and guarantees fair and consistent monitoring over modern issues of academic integrity.

The main research question addressed in this study is: How suspicious facial movements of students and banned objects during examination can be tracked through reel-time feed using python libraries OpenCV, TensorFlow and YOLO?

Another objective here is the behavior is categorized based on how likely the behavior is to indicate cheating. Clear violations are listed as serious actions like detecting a mobile phone as in Figure 1. or another person helping the student. Less definitive behaviors include ones like frequent eye movements, yawning, or head movements that may be to glance at materials they shouldn't be looking at or not which can be observed in Figure 2. Voice detection also alerts if someone is talking or making any unordinary sounds, meaning the victim may need help from outside. By adopting this approach, we can monitor very strictly without getting false alarms on fair examination supervision.

Figure 1. Clear violation.

Figure 2. Less definitive behavior.

This paper is structured as follows: Section 2 reviews related work on facial detection and object tracking. Section 3 outlines the methodology and tools used. Section 4 covers the system design, while Section 5 details its implementation. Section 6 evaluates the system's performance, and Section 7 concludes with key findings and future improvements.

# 2 Related Work

## 2.1 Face Detection and Recognition

This paper focuses on the principle of face recognition which can be explained by splitting the image into two parts. The first of which includes faces and the second of which does not, and the results are being detected. To measure the performance of the given number of faces present. The study has indicated that OpenCV and python are accurate in face detection. Since recognition is one of the main tasks in computer vision, detection is essential. This paper contain several works on face detection using OpenCV for example, real-time human head tracking, recognizing human emotions, and face mask detection. The advantages of face recognition are outlined, some of which are pointed out to be in use in such areas as schools to track students, and tracking of students through registration (Hasan, R.T. and Sallow, A.B., 2021).

The author brings to light the significance of record keeping of student’s performance through exams and ensures that rules are put in place with a view to discourage cheating. Remote assessments integral to online learning being adopted by pandemic have exposed issues of monitoring students during such assessments. In order to address this, the study suggests a system based on Eigenfaces for feature extraction and Support Vector Machines (SVM) for classification using real time face recognition for higher accuracy. The system operates on the OLR dataset of 400 entities, although its diversity is restricted. The paper points out drawbacks of the algorithm and provides alternatives such as LBPH, Fisher faces, SIFT, SURF etc. The extracting embedding, SVR classifier training, embedding storing to pickle file for faster retrieval is the training process. Nevertheless, results indicate that we are able to capture the performance of the student with improved accuracy as the dataset of functions to evaluate becomes larger, resulting in improved student monitoring. Further work could use other techniques, such as CNNs, for better performance in different conditions in order to improve upon online exam monitoring systems (Geetha, M., Latha, R.S., Nivetha, S.K., Hariprasath, S., Gowtham, S. and Deepak, C.S).

This study proposes a facial recognition system for attendance recording using a Multi-Layer Perceptron (MLP) algorithm and Python libraries. Images are captured via a camera, stored, and matched with a database to mark attendance in an Excel sheet. OpenCV, a multi-language library for real-time computer vision, supports functions like face detection and object recognition and is compatible with various operating systems. The method includes reading images, extracting facial encodings, and training an MLP classifier with one hidden layer of 100 neurons. OpenCV is integrated with face recognition libraries to detect faces, recognize them, and record attendance with timestamps. The study demonstrates an efficient and accurate attendance system that is easy to develop using Python and MLP (Ali, A.K. and Mustafa, J.Y., 2024).

## 2.3 Comprehensive Proctoring Systems

This paper presents ProctorXam, a proctoring application designed to detect integrity problems in online exams. ProctorXam, uses real-time video analysis to detect emotions such as body posture, facial expressions, muscle activity, head orientation. The system ensures that instructors will be confident to administer closed book exams, while simultaneously alerting students if suspicious activity is detected. The dataset consists of 130,000 frames of 20 participants with different head movement control is used for the study. The training set of the head posture module provides a comprehensive test bed. The performance and credibility of ProctorXam's head pose estimation is compared with Dlib and 3DFFA. At the moment, the system is running under low FPS and will be optimized in the future for better efficiency (Ashwini, S., Kumar, B. and Sathya, V., 2024).

In this paper, we propose a rule based inference system to detect cheating during online exams, using captured active windows, video and audio recordings. Capture of active windows logs all processes on the examinee’s device – including unauthorized software, browser usage as well as USB connection. The student’s face is monitored through video input to attend class and to misbehave, for which facial feature detection, head pose estimation and yawn angle tracking with a 15° cheat detection threshold is used. Additional individuals are identified via multi-face detection. Noise levels and variations in inferred side conversations are checked by audio analysis. Disturbances are detected based on system yawn angle changes combined with audio levels and window activity. We tested on a dataset of staged cheating scenarios where we were successful in detecting cheating with minimal false positives equal to that of manual proctoring. The result, a system using a multi model approach, exemplifies the system's potential for automated online exam proctoring (Prathish, S. and Bijlani, K).

Using facial recognition and live proctoring, the developed exam management system improves online exams. It helps identify students by faces, monitor any suspicious action, and attend them, decreasing manpower in record keeping. The system includes key modules: It comprises modules for enrollment and identification registration, live proctoring to avoid cheating and an attendance module to record attendance. It's built with Flask for the client server communication, React for the server communication, and DeepFace for facial recognition. To improve recognition, it is important to create a face matching feature vector with VGG face model, particularly VGG-19, which may overfit training data. However, problems such as lighting variations or the presence of accessories like mask or glasses, coupled with the overfitting to these problems, may manifest as false positives or negatives, which adversely affect the real time performance of such systems. Future research would focus on accuracy and flexibility of models like FaceNet (Rajalakshmi, B., Dandu, V.K., Tallapalli, S.L. and Karanwal, H).

The research suggests a system that can improve control, tracing, and the recognition of students, especially to attend classes. In this manner, it will be easier for the educators to attend to the issues of absence in a proper time and such rules such as having to be present for certain lessons within students. Also, the current system does not have a functionality of sending personalized notification to a particular individual, which is crucial in security purposes. The paper also suggests that the real time tracking, and notification should also be included in this case in a bid to improve on the security of the system. Of all the components, OpenCV is most crucial for live stream playback. Also, it is possible to apply face detection models to the surveillance of large crowds. The architecture section also discusses how YOLO was included into the face detection model and the significant advantages of it is useful to do so when working with big data sets. This paper also reveals that the conventional methods are not helpful. When it is almost impossible to classify the objects or when there is an increase in the number of objects in a scene, while YOLO has a high accuracy in real conditions or conditions with multiple objects (Tejas, V.H., Harshitha, S., Yogitha, R., Kalaiarasi, G. and Selvi, M., 2024).

## 2.3 Object Detection and YOLO-based Techniques

This paper points out that in the last couple of years, small object detection has been gaining popularity and has attracted the attention due to advancements made in Optical Remote Sensing techniques that have application in traffic. These are the management and military spying. Some of the more recent advanced YOLO models include TPH-YOLO, FE-YOLO and CA-YOLO have also been presented for remote sensing images. The paper uses the transformer encoder blocks in TPH-YOLO to enhance the object feature while the deformable convolution is used in FE to handle the variation of objects. YOLO for feature fusion. The study finds that FFCA YOLO and the reduced version of L-FFCA-YOLO are feasible to use techniques in small target detection in remote sensing images. The proposed modules enhance feature representation and background suppression and the new dataset USOD can be useful for further study. There are so many versions of YOLO each and every research finding proposes a different version for object detection (Zhang, Y., Ye, M., Zhu, G., Liu, Y., Guo, P. and Yan, J., 2024).

The proposed method focuses on speed and accuracy of the face detection and verification in order to achieve effective and reliable online exam monitoring. Viola Jones, LBP, MTCNN, and YOLO face are face detection methods, and YOLO face outperforms in detection of facial poses, illumination variations and high accuracy with high scalability. In face verification, we compare identified faces using FaceNet, which embeds facial images into Euclidean space with over 90% accuracy and little hyperparameter tuning. Because FaceNet is robust for image compression, it can be deployed on mobile platforms. Evaluation parameters are detection times, accuracy, dataset size and training times, and preprocessing is done using TensorFlow and OpenCV. While the system has several advantages, it must contend with the physical hardware limitations of mobile devices as well as potential risks of making use of uni-modal biometrics. This work could also be integrated with a multimodal system to reach higher accuracy in various conditions (Ganidisastra, A.H.S).

### 2.4 Gap in Existing Applications

There are many applications for exam monitoring and proctoring. However, there remains a strong gap in systems which effectively combine face recognition, object detection and voice detection utilizing the latest advancements in these fields. Existing solutions mostly address one or two modalities, ignoring the synergy of merging those to have a final monitoring system. This warrants an updated methodology utilizing these new technologies simultaneously as collaboration in order to enable a robust and reliable exam surveillance.

## 3 Research Methodology

First, an extensive problem analysis was conducted to identify the shortcomings of existing monitoring systems and the lack of integrated solutions that offer face recognition, object detection and audio analysis. The approach leverages advanced computer vision and audio processing technologies to detect suspicious behaviours occurring during examinations in a manner which is both methodologically rigorous and scientifically valid. Based on this, the study defined its primary objective: for the purpose of developing an automation system that uses advanced technologies to track facial movements, detect prohibited items and analyse audio signals to monitor and closely monitor students behaviour during exams. In order to fill these gaps, the solution proposed was to combine the use of state-of-the-art tools and frameworks to allow learners to upload their assessments to cloud storage, which next they will have to retrieve to access their score and feedback. The high-end performance computing environment that has been used on the research using AMD Ryzen 7 6800HS with Nvidia GeForce RTX 3050 GPU and 16 GB DDR5 RAM, running the python IDE with the libraries installed such as Scikit-Learn, TensorFlow/Keras, opencv, NumPy, sounddevice, flask and os.

### 3.1 Data Preprocessing

Real-time monitoring system accuracy and efficiency depends highly on the preprocessing of input data for detection and analysis purposes. The system works on both visual and auditory data streams and uses several preprocessing techniques specifically designed to them. Reprocessing techniques tailored to their specific requirements. For visual data captured from the webcam, each frame is resized to standardized dimensions, for instances 416x416 for YOLO to align with model input requirements, reducing computational overhead and maintaining compatibility with pre-trained models (Thota, J.R. and Padala, A., 2024.). Between these two image crops, the frames also undergo color space conversions to grayscale for reduced computation during facial landmark detection and YCrCb and LUV for spoofing detection involving distinctions in face foregrounds (Ghosh, T. and Naskar, R., 2024). Image noise is reduced with techniques in the form of Gaussian blurring and median filtering to remove damaged pixels while retaining important details. In order to perform object detection on YOLO, they are converted into normalized input blobs so they are consistently processed as efficiently as possible (Thota, J.R. and Padala, A., 2024.). Furthermore, they crop and align faces that are detected.

Real time audio data is captured using microphone, preprocessed in such a way that the analysis of speech and noise is accurate. For real time processing, the audio stream is sampled at a rate of 16 kHz and segmented into 1024 frames per segment. Audio input is analyzed to determine the root mean square (RMS) energy measured from spikes of excitation that separates silence, speech, and background noise. Ambient sounds are filtered out by applying a silence threshold, all speech noises are isolated for further analysis. Collectively, these preprocessing steps improve the detection model accuracy, decrease computational load, and increase feature extraction to provide a robust and computationally efficient monitoring system for effective examination environment tracking.

## 3.2 Key Detection Models and Their Roles

### 3.2.1 ResNet10 SSD (Single Shot Multibox Detector) for Face Detection

The proposed system uses a ResNet10 SSD as its primary model for face detection. This model is lightweight convolutional neural network implemented using OpenCV's DNN module and pre-trained using the Caffe framework. It uses the SSD architecture, that learns to predict object bounding boxes and class probabilities in a single shot. OpenCV is used to process video frames with the system isolating faces in the feed and passing the resulting regions along to other modules to be further analysed by creating a face detection model with pre trained weights made by cv2.dnn on Caffe Resnet10 iterating 140000 times with parameters 'deploy.model' which has the model architecture to be utilized later for its deployment as shown in Figure 2 (Aggarwal, G., Sinha, A., Srivastava, P. and Agarwal, P., 2021).With ResNet10 SSD, we are able to detect faces under various scenarios with a high robustness, including variable lighting and multiple facial orientations.

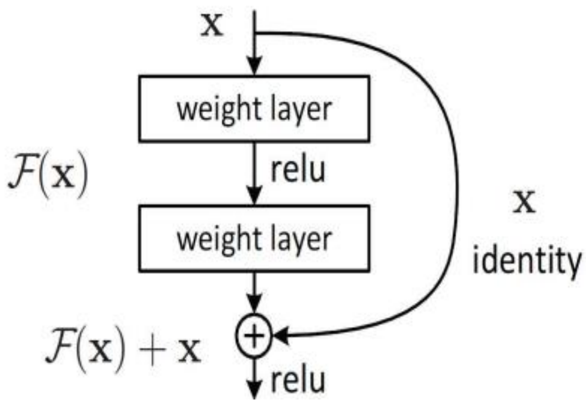


Figure 3. ResNet Building block.

### 3.2.2 TensorFlow for Facial Landmark Detection

To detect and track facial landmarks, we use TensorFlow so that we can track mouth opening movements in the system. The model used to identify points identifies key landmark points used in people detection such as the eyes, the nose and the mouth which are important when detecting suspicious activities which include frequent whispering. In real time, ResNet10 SSD module acts as backbone to detect faces. Being a deep learning framework, TensorFlow can track these very subtle facial movements quite precisely even in dynamic surroundings (Thota, J.R. and Padala, A., 2024). This is a necessary feature to help catch nonverbal hints of cheating attempts, but only among so much natural behaviour such as yawning or head tilts to avoid false positives.

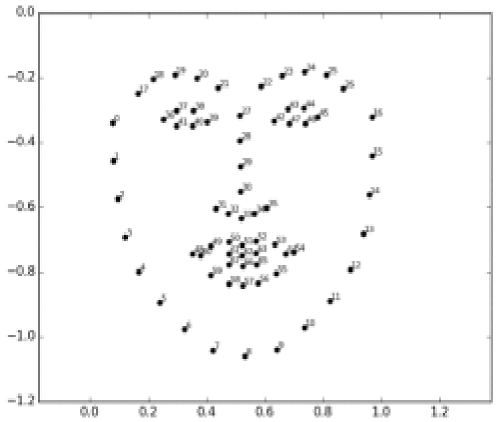



Figure 4. The facial landmarks.

### 3.2.3 YOLO (You Only Look Once) for Object Detection

In the examination environment, the YOLO object detection framework identifies prohibited objects like mobile phones in real time. YOLO divides input images into grid cells, predicting bounding boxes and class probabilities in a single forward pass, enabling rapid and efficient multi-object detection. For this project, YOLO is fine-tuned with pre-trained COCO dataset weights to detect specific classes, such as person and cell phone. These weights enhance the model's ability to recognize objects by their perceptual features. When a mobile phone is detected, the system logs the event. YOLO excels in handling cluttered scenes and diverse environments, making it ideal for exam settings with distractions. It balances speed and accuracy, ensuring effective banned object detection without compromising the monitoring system’s performance or integrity (Nguyen, H.V., Bae, J.H., Lee, Y.E., Lee, H.S. and Kwon, K.R., 2022).

### 3.2.4 Sounddevice for Voice Detection

Audio input is monitored with the sounddevice library, looking for speech or odd noises indicative of having to ask for help further. In this module we capture and process real-time audio continuously in a background thread. The speech or noise disturbances are identified in the audio samples by the system by comparing the Root Mean Square energy of the audio samples to a predefined silence threshold real-time audio using a background thread (Tuunanen, T., 2020). When speech or other important noises are picked up, we tell them they’re a potential sign of cheating. The addition of this voice detection module provides the system with additional monitoring capabilities beyond the use of visual data, by adding an auditory analysis component to its monitoring efforts, resulting in multi modal exam surveillance.

### 3.2.5 scikit-learn for Face Spoofing Detection

In order to ensure the integrity of the system, face spoofing detection using scikit learn is used to detect when someone attempts to bypass the system with a fake face by photo or video. The module utilizes a pre trained classifier that feed with the extracted features from some color spaces like YCrCb and LUV, to classify the real face and spoofed one. The classifier is applied to the color histograms computed from regions containing detected face (He, J. and Luo, J., 2019). This leads to a greater protection of the security and reliability of the monitoring system as only genuine faces will be monitored.

### 3.3 Statistical Analysis

Statistics of the system is analysed for performance and efficiency of its detection modules, as evaluated by their key metrics for validations. The system calculates precision for mobile and person detection using YOLO to be the accuracy of identifying banned objects such as mobile phones using true positives and false positives (Nguyen, H.V., Bae, J.H., Lee, Y.E., Lee, H.S. and Kwon, K.R., 2022). An analysis of frames per second (FPS) is used to determine that the system meets real time processing requirements, while detection times are logged and plotted vs. frames to determine performance bottlenecks. Precision curves and FPS trends are used to evaluate system reliability at different conditions. The mouth opening detection module counts the number of open versus closed frames, and the average duration of detected mouth open events. Behaviours are analysed across time through time series plots, plotting the status of the mouth over time, and histograms of mouth open durations. All modules are tested using a controlled video input with performance metrics of precision, recall and detection consistency recorded. Audio analysis and face spoofing are validated with manual UI alerts since no automated testing integration exists. This analysis showcases the robustness of the system as well as its areas for optimization and makes for a scientifically validated monitoring solution.

# 4 Design Specification

## 4.1 System Architecture

The system is built on a modular architecture, comprising four primary components: Based on face detection, facial landmark tracking, object detection and voice detection. In this, we use the face detection module based on the ResNet10 SSD created using OpenCV's DNN module to find and separate face regions out of the video feed. Next, the facial landmark module is used to track the facial key landmarks (68) namely, the eyes, the mouth, and nose after detecting these face regions using TensorFlow, which can be used to detect behaviour related to mouth opening, amongst others. At the same time, the object detection module uses YOLO for the detection of prohibited items such as mobile phones in the video feed. The voice detection module, implemented using the sounddevice library, is running in parallel and listens to audio input to detect speech or any unusual sounds and moreover flags on UI. These components are incorporated into a Flask based backend enabled for real time monitoring and analysis.

## 4.2 Workflow

The system takes in input through a series of steps. Next, a webcam is used to capture video input, and a microphone is to capture an associated audio stream. Faces are detected and facial landmarks are tracked with processing of the visual input, and objects and speech are recognized with the audio input. All detection modules log the results in real time and display them for real time flagging of suspicious behaviours. Critical events like the detection of a banned object and if there is more than one person the application saves the image in the database and generates alert to admin.

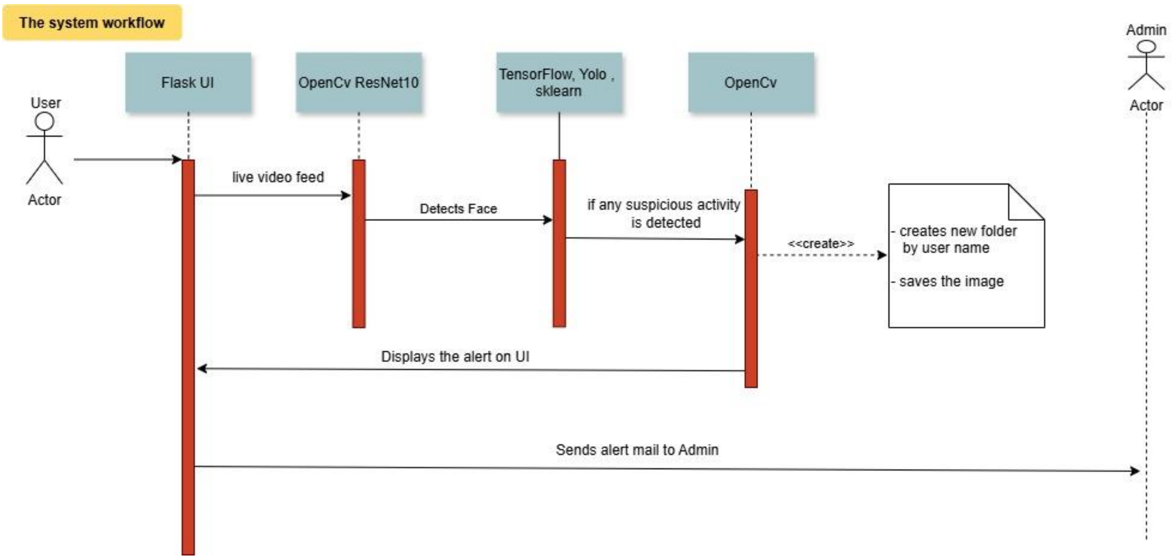


Figure 5. The design steps.

## 4.3 Component Interaction

In practice, the components work synchronously to ensure their entire monitoring. The first visual processing pathway is the face detection module which isolates regions of objects within the frame and feeds them into the facial landmark detection module. The second part of the video feed, the object detection module checks for banned items, without interacting with the other part. At the same time, the voice detection module scans audio stream to detect speech or noise. These modules are aggregated in a Flask backend that presents them to the end user in a consolidated interface for review.

## 4.4 User Interface

The system consists of a Flask powered web interface that is simple to use. Next, the real-time video feed is overlaid with results from detection and logs of flagged behaviors and objects are displayed through this interface. If there is a strong suspious behavior, administrators will be able to login and see what images the system captured.

## 4.5 Scalability and Performance

The system is built for real time processing of video and audio and already uses threading library for handling simultaneous processing of video and audio. The design of each module allows for minimal latency while still maintaining efficient operation under the constraint of resources. The architecture is modular so that it can be scaled up by adding modules for advanced behavior analysis or other extended object detection capabilities.

# 5 Implementation

## 5.1 System Outputs

Face Detection and Facial Landmark Analysis: ResNet10 SSD model is used to detect faces on video frames processed by the system which creates bounding boxes around detected faces. For further analysis, these bounding boxes are passed. TensorFlow is then used to analyze detected face regions to identify 68 regular landmarks of the face. These points are overlaid on the live video feed to mouth activity movements. Suspicious behaviour like repeated movement of the opening of the mouth, is flagged for attention.

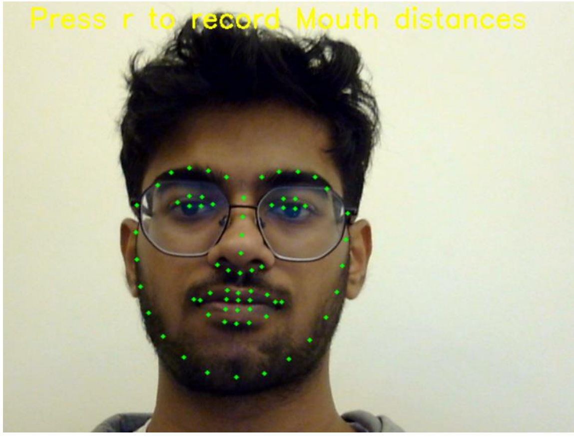



Figure 6. Facial landmarks implentation.

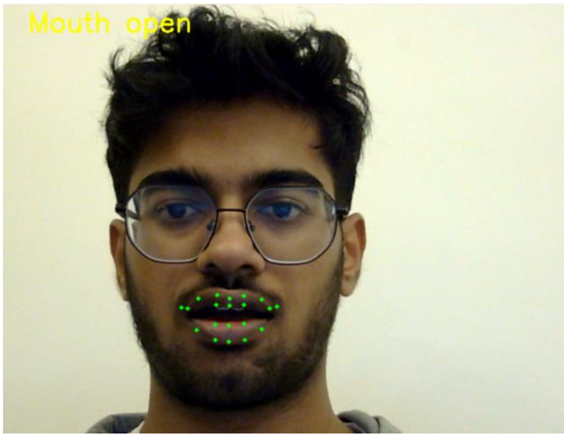



Figure 7. Mouth activity.

Object Detection: Within video frames, YOLO identifies prohibited items, like mobile phones. Class labels and confidence scores of the detected objects are logged. For banned items, generation of alerts is done, in the dataset directory.

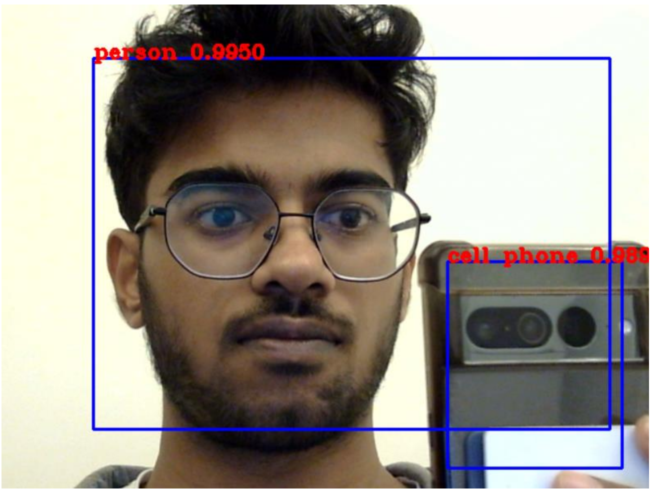



Figure 8. Implementation of Object and person detection.

Face Spoofing Detection: The system uses scikit-learn to differentiate between real and spoofed faces. This is accomplished by feature analysis of the features extracted from specific color spaces, which were YCrCb and LUV. Spoofing attempts are flagged to ensure the integrity of the monitoring process.

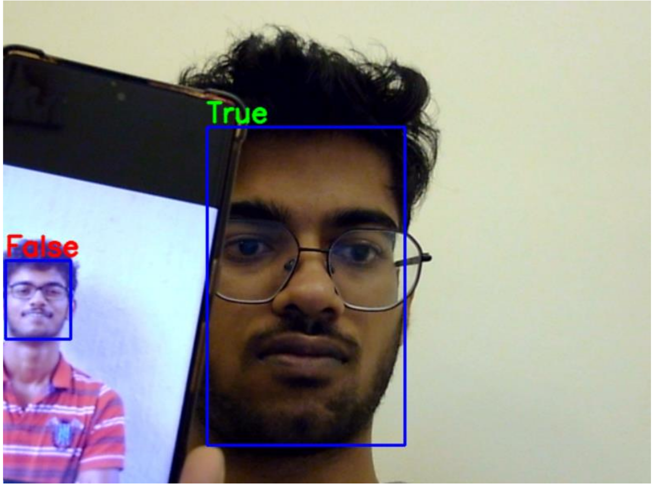


Figure 9. Implementation of face spoofing.

Voice Analysis: Speech and unusual noises are detected on audio inputs. If there is speech more than a predefined threshold, an alert is generated, and then the event is logged along with timestamps. Integrated

Logs and Alerts: Logs of flagged events, including detection events such as timing, detection type such mobile phone detected or secondary person in the frame and relevant snapshots for supporting context, are compiled into the system. A web interface is used to present these outputs.

### 5.2 System Integration and Presentation

The solution leverages face detection, landmark analysis, object detection, face spoofing detection and voice analysis into a unified pipeline for real time monitoring. The central interface is Flask, built using Python 3.7 and Python IDLE, and does the data routing, and presents results on a web-based platform with HTML, CSS for UI components. Simultaneous audio and video streams process via multithreading, displaying the results as a video live feed that shows four frames. These events can be flagged with a snapshot, detailed description, and placed in an administrator dashboard where you can manage your users and review the events. Annotated images, detection logs, and processed frames are confirmed as outputs to be used for a reliable deployment at examination sites. SQLite3, OpenCV2, TensorFlow Keras, NumPy, OS, Threading, Flask are the main libraries have been used to obtain robust performance.

## 6 Evaluation

A detailed analysis of the system's performance is presented here, with key findings and their implications discussed. Efficiency of the implemented modules are evaluated using statistical tools to analyse the results in order to fulfil the system objectives. The precision score of YOLO module for detecting prohibited objects which are mobile phones, was 1.0 in testing. This means each detection was correct and we logged no false positives. The results show that the system's high precision reliability can distinguish mobile phones in the examination environment,

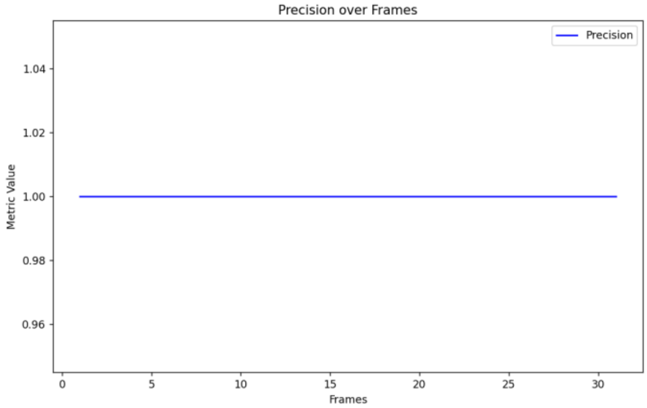



Figure 10. Precision of YOLO model.

Evaluation period of recording mouth activity over 13 seconds, it recorded 40 times up and down ‘mouth open’ status and 49 times up and down ‘mouth closed’ status. Average duration for which the mouth was open was 0.60 seconds. The system is able to track and log mouth movement from fully open mouth to whispering.

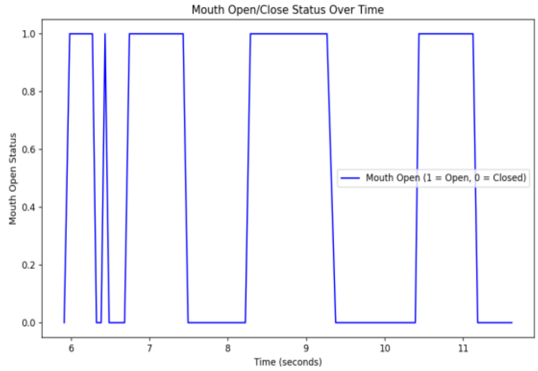



Figure 11. Tracking mouth movements.

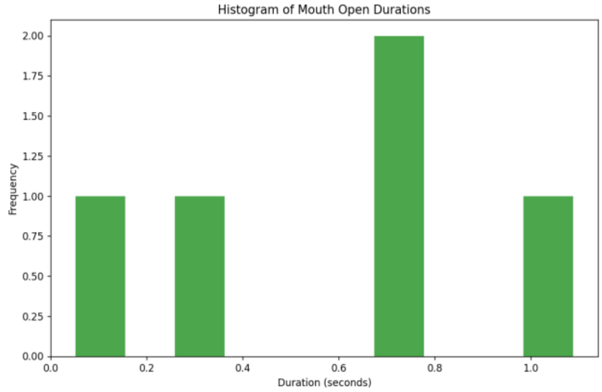



Figure 12. Frequency of mouth open duration.

An automation script was run to simulate real world conditions by processing every frame of a test video to evaluate the system as a whole. The script evaluated the face detection, facial landmark analysis, and object detection under controlled conditions integration and performance of all modules. The system could process all frames without errors to prove its robustness and demonstrated the smooth interplay of the parts.

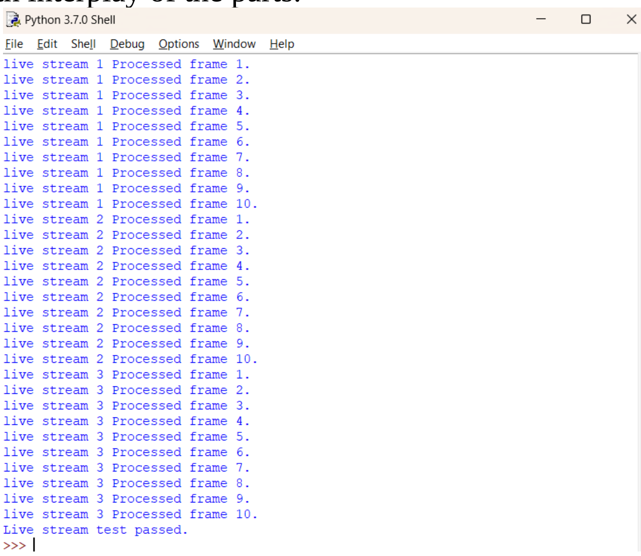



Figure 13. Passed autmation testcases

The application system which was developed with the purpose of monitoring the exams with the help of technologies. Figure 14. The Index Page serves as the starting point, offering two options: Student Login and Admin Login. Students or User type can login or the signup option to get into the examination system.

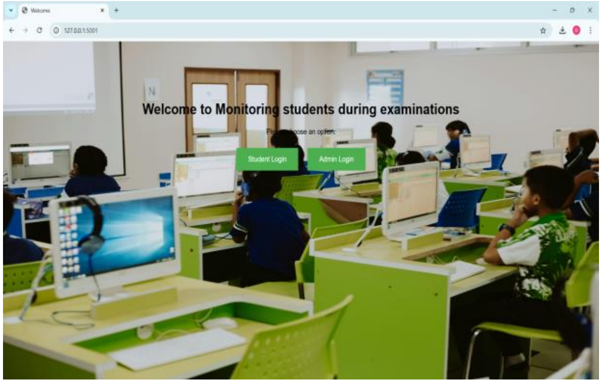



Figure 14. Index Page.

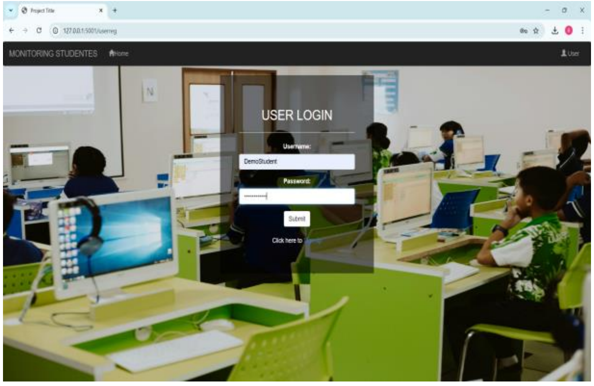



Figure 15. Studet login page.

After students have logged in, they are automatically taken to the Examination Page. The Figure 16. Shows Examination Page has built-in real-time monitoring options. Four video frames are featured on the page which allows to constantly monitor the student's surroundings. The first frame deals with mouth open detection which is achieved by using the facial landmarks if any sort of mouth activity is noticed. The second frame uniquely focuses on spoofing detection, here the system works on the live feed discrepancy to identify the face spoofing which can be done through held photos or videos. In the third frame, it detects if there is one person and the fourth frame targets prohibited items including Mobile phone. Further, a speech detection module is used to detect any activity within the audio stream and triggers an alarm if speech levels go beyond a particular limit. In addition to these monitoring functions there is a question panel in front of the students in which they can take their tests and send in their answers.

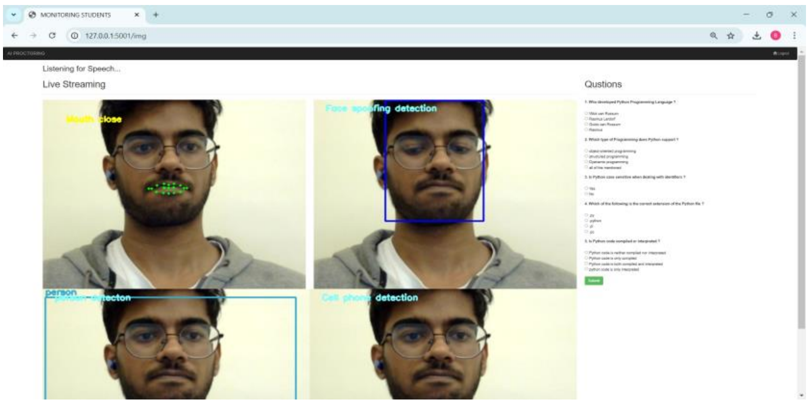



Figure 16. Examination Page.

If there is any form of violation during the exam like usage of a mobile phone in the exam area, speech detection or face spoofing the system flags it on UI as shown in Figure 17. and takes annotated snapshots for future use. The system also generates an alert email to the administrator and a possible violation accompanied by a record of the violation shown in Figure 18.

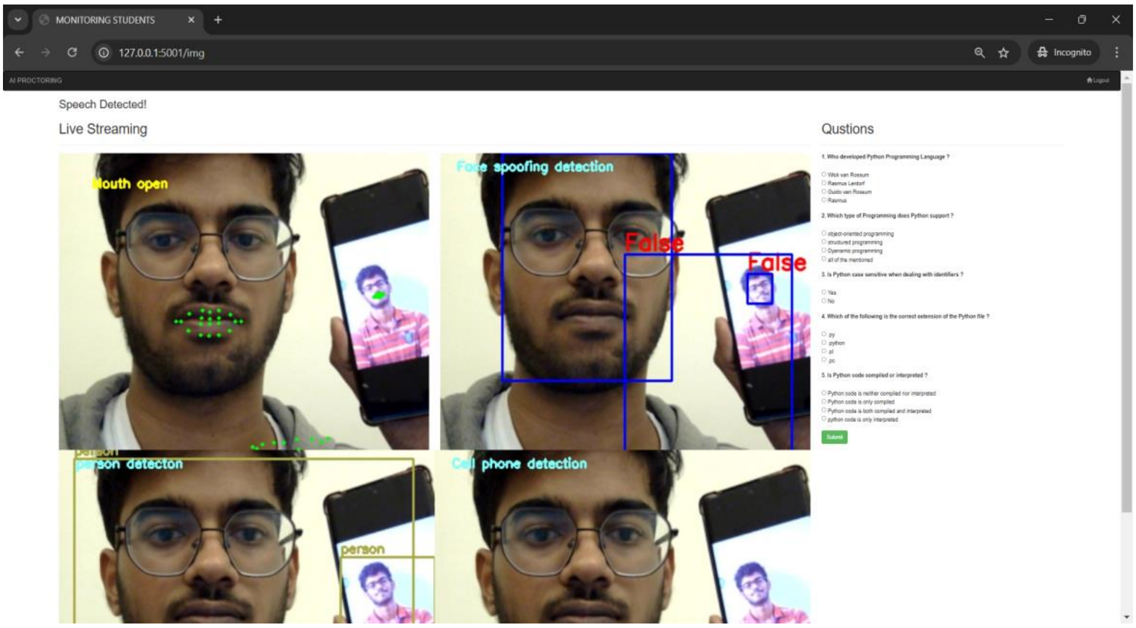



Figure 17. All frames detecting violation and flagging it.

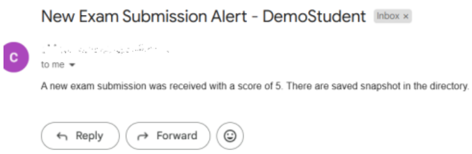
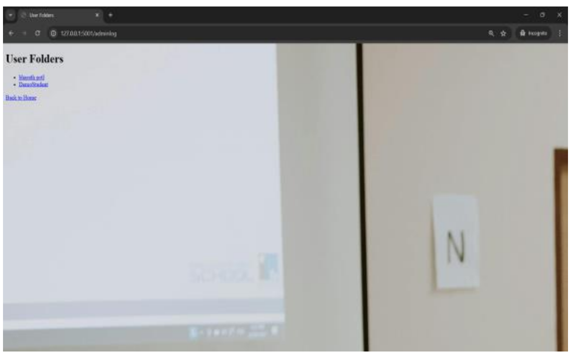



Figure 18. Alert Mail sent to admin.

The saved snapshots can be accessed from the ‘Admin Login’ to enable administrators to make decisions. They can view comprehensive transcripts including snapshots, timestamp and description of the delaying behaviors which can be observed in Figure 19 and 20. This efficient mechanism allows administrators to deal with violations appropriately and such things like warning or nullification of examinations appropriately. These features of the system’s interface also guarantee students and administrators’ engagement in communication, providing a stable and effective monitoring system.



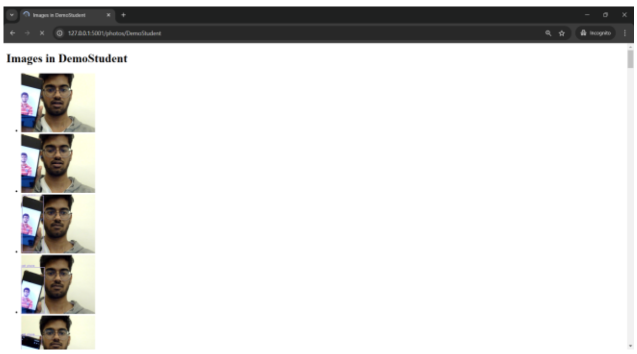




Figure 19. Snapshot folders of different students.

Figure 20. Snapshots of a student.

### 6.1 Discussion

In this study, advanced techniques are integrated into an automated monitoring system that aims to improve the security of examination environments. Even though the spoofing detection module effectively identifies an effort to use photos or video of some other person, more advanced 3D face models or liveness detection algorithms could improve accuracy against more sophisticated spoofing. The audio activity flag created by the speech detection module, sometimes erroneously counted environmental noise as violations. Through noise cancelling preprocessing or training on large volume audio datasets, its precision can be improved Facial landmark analysis recorded 24 mouth openings, 79 closings, and an average open duration of 0.68 seconds, tracking behaviour with some false positives from head movements or posture adjustments. Hence, embedding contextual rules or applying attention-based models, further focus can be put on meaningful behaviours. A main limitation was the use of static, controlled environments for testing, which does not capture real exam settings in terms of varying lighting, backgrounds and device quality.

## 7 Conclusion and Future Work

The objective of this study was to track suspicions facial movements and banned objects during exams using Python libraries of OpenCV, TensorFlow and YOLO. A monitoring system successfully integrated facial landmark analysis, object detection, and audio monitoring, with functionalities like mouth movement detection, face spoofing detection, looped object tracking, speech analysis, saving the snapshots and all presented through a Flask based interface. Key results consist of a precision score of 1.0 and successful analysis of 24 mouth openings, 79 mouth closures and average open duration of 0.68 seconds. Frame by frame processing was confirmed to work reliably by automation tests, and violations were properly logged and sent to an administration panel. Some limitations are the 3.3 FPS rate, which may not be suitable for high number of granularities, and speech detection needing advance natural language processing to capture effective context. In addition, face spoofing detection could also be leveraged against techniques such as high-resolution images.

Future work can concentrate on improving the FPS rate to allow for smoother and more detailed monitoring and optimizing model performance for faster processing while improving spoofing detection through liveness analysis and expanding audio monitoring with phrases used to cheat. Secondly, adapting the system to accommodate multiple exam sessions would also make the system more practically useful. Lastly by using the detected facial landmarks extending the tracking capabilities of eye and head orientation to detect suspicious activity. Having had these enhancements, the system lives up to its commercial potential to serve as a secure system to monitor online students.

## References

Ashwini, S., Kumar, B. and Sathya, V., 2024, May. ProctorXam-Online Exam Proctoring Tool. In 2024 International Conference on Electronics, Computing, Communication and Control Technology (ICECCC) (pp. 16). IEEE.

Tejas, V.H., Harshitha, S., Yogitha, R., Kalaiarasi, G. and Selvi, M., 2024, May. Tracking A Student Using Face Recognition. In 2024 International Conference on Intelligent Systems for Cybersecurity (ISCS) (pp. 16). IEEE.

Rajalakshmi, B., Dandu, V.K., Tallapalli, S.L. and Karanwal, H., 2023, August. ACE: Automated Exam Control and E-Proctoring System Using Deep Face Recognition. In 2023 International Conference on Circuit Power and Computing Technologies (ICCPCT) (pp. 301-306). IEEE.

Hasan, R.T. and Sallow, A.B., 2021. Face detection and recognition using opencv. Journal of Soft Computing and Data Mining, 2(2), pp.86-97.

Ali, A.K. and Mustafa, J.Y., 2024. User-Friendly Interface Attendance System Based on Python Libraries and Deep Learning. International Journal of Data Science, 5(1), pp.56-62.

Geetha, M., Latha, R.S., Nivetha, S.K., Hariprasath, S., Gowtham, S. and Deepak, C.S., 2021, January. Design of face detection and recognition system to monitor students during online examinations using Machine Learning algorithms. In 2021 international conference on computer communication and informatics (ICCCI) (pp. 1-4). IEEE.

Prathish, S. and Bijlani, K., 2016, August. An intelligent system for online exam monitoring. In 2016 International conference on information science (ICIS) (pp. 138-143). IEEE.

Zhang, Y., Ye, M., Zhu, G., Liu, Y., Guo, P. and Yan, J., 2024. FFCA-YOLO for small object detection in remote sensing images. IEEE Transactions on Geoscience and Remote Sensing.

Ganidisastra, A.H.S., 2021, April. An incremental training on deep learning face recognition for m-learning online exam proctoring. In 2021 IEEE asia pacific conference on wireless and mobile (APWiMob) (pp. 213-219). IEEE.

Farkhod, A., Abdusalomov, A.B., Mukhiddinov, M. and Cho, Y.I., 2022. Development of real-time landmark-based emotion recognition CNN for masked faces. *Sensors*, *22*(22), p.8704.

Jia, J. and He, Y., 2022. The design, implementation and pilot application of an intelligent online proctoring system for online exams. *Interactive Technology and Smart Education*, *19*(1), pp.112-120.

Hettiarachchi, S., 2021. Analysis of different face detection andrecognition models for Android.

Thota, J.R. and Padala, A., 2024. Human Remains Detection in Natural Disasters using YOLO: A Deep Learning Approach. *Engineering, Technology & Applied Science Research*, *14*(6), pp.17678-17682.

Ghosh, T. and Naskar, R., 2024. Less is more: A minimalist approach to robust GAN-generated face detection. *Pattern Recognition Letters*, *179*, pp.185-191.

Aggarwal, G., Sinha, A., Srivastava, P. and Agarwal, P., 2021, November. Identification of Unmasked Faces: Protection in Public Places from COVID-19 and Other Infections. In *2021 International Conference on Technological Advancements and Innovations (ICTAI)* (pp. 637-642). IEEE.

Gwak, M., Vatanparvar, K., Kuang, J. and Gao, A., 2022, July. Motion-based respiratory rate estimation with motion artifact removal using video of face and upper body. In *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)* (pp. 1961-1967). IEEE.

He, J. and Luo, J., 2019, July. Face Spoofing Detection Based on Combining Different Color Space Models. In *2019 IEEE 4th International Conference on Image, Vision and Computing (ICIVC)* (pp. 523-528). IEEE.

Fergus, P. and Chalmers, C., 2022. Deploying and hosting machine learning models. In *Applied Deep Learning: Tools, Techniques, and Implementation* (pp. 299-317). Cham: Springer International Publishing.

Liu, C., Tao, Y., Liang, J., Li, K. and Chen, Y., 2018, December. Object detection based on YOLO network. In *2018 IEEE 4th information technology and mechatronics engineering conference (ITOEC)* (pp. 799-803). IEEE.

Nguyen, H.V., Bae, J.H., Lee, Y.E., Lee, H.S. and Kwon, K.R., 2022. Comparison of pre-trained YOLO models on steel surface defects detector based on transfer learning with GPU-based embedded devices. *Sensors*, *22*(24), p.9926.

Tuunanen, T., 2020. *Real-time sound event detection with python* (Master's thesis)